求解并行机拖期与能耗成本优化调度的混合教—学算法 *

王永琦，吴　飞，江潇潇，王春媛

(上海工程技术大学 电子电气工程学院，上海 201620)

摘　要：针对加工时间可控的并行机调度，提出了一类考虑拖期与能耗成本优化的调度问题。首先对调度问题进行了问题描述，并建立了整数线性规划模型以便于 CPLEX 求解。为了快速获得问题的满意解，提出了一种混合教-学算法。结合问题的性质，设计了编码与解码方法以克服标准教-学算法无法直接适用于离散问题的缺点。同时，构建了基于变邻域搜索的局部搜索算子以强化混合算法的搜索性能。最后，对加工时间可控的并行机调度问题进行了仿真实验，测试结果验证了本文构建的整数线性规划模型和混合算法的可行性和有效性。

关键词：并行机调度；拖期；能耗；可控加工时间；教-学优化算法

中图分类号：TP301.6　　doi: 10.3969/j.issn.1001-3695.2017.09.0929

Hybrid teaching-learning-based optimization algorithm for optimizing tardiness and energy cost on parallel machine scheduling

Wang Yongqi, Wu Fei, Jiang Xiaoxiao, Wang Chunyuan

(*School of Electronic & Electrical Engineering*, *Shanghai University of Engineering*, *Shanghai* 201620, *China*)

Abstract: For the parallel machine scheduling with controllable processing times (PMS-CPT) , this paper proposed a scheduling problem for optimizing tardiness and energy cost. First, this article developed an integer linear programming (ILP) model for the proposed scheduling problem in order to facilitate the CPLEX solver. Second, this paper established a hybrid teaching-learning-based optimization (HTLBO) algorithm to quickly obtain satisfactory solutions. This algorithm adopted a new coding and decoding method according to the nature of the proposed parallel machine scheduling problem, which overcame the weakness of standard teaching-learning-based optimization algorithm that could not be directly applied to discrete problems. Meanwhile, a local optimizer was proposed based on variable neighborhood search (VNS) , which aimed to enhance the performance of the hybrid algorithm. Finally, the simulations were conducted to solve instances of PMS-CPT. The experimental results demonstrate the feasibility and validity of the proposed ILP model and hybrid algorithm.

Key Words: Parallel machine scheduling; Tardiness; Energy; Controllable processing times; Teaching-learning-based optimization algorithm

0　引言

作为当前实际生产制造过程中的一类典型的调度问题，并行机调度(parallel machine scheduling, PMS)问题具有相当广泛的应用背景，包括云计算、半导体加工、汽车制造等行业[1]。传统的 PMS 问题致力于优化与作业完成时间相关的指标，包括最大完工时间、提前期/拖期等。随着客户定制化的快速发展以及全球能源价格的攀升，准时化与节能调度成为当前车间调度的重要方面[2-4]。因此，进行并行机拖期与能耗成本优化调度方面的研究具有重要的理论意义和工程应用价值。

PMS 问题已被证明为 NP-hard 问题，快速、高效的算法设计已经成为这一领域的重要课题。相关的求解算法可以归纳为以下三类：精确算法、启发式算法和智能算法[5]。精确求解算法包括动态规划、分枝定界等算法，这类算法能够获得 PMS 问题的准确解，但随着问题规模的扩大，这类算法所需的计算时间将急剧增长。启发式算法利用调度规则来实现加工任务在不同

基金项目：国家自然科学基金资助项目：多核实时约束下基于 DVFS/DPM 节能技术的任务调度研究（F020207）；上海市工程技术大学《信号与系统》平台课程建设（k201602004）

作者简介：王永琦（1977-），女，河南省焦作市，副教授，博士研究生，主要研究方向为智能计算与信号处理（wangyongqi17008@163.com）；吴飞（1968-），男，山东，教授，博士，主要研究方向为计算机并行处理与节能控制；孙建华（1978-），女，河南省焦作市，副教授，博士，主要研究方向为网络安全和信息安全，入侵检测.**基金项目：**国家自然科学基金资助项目（F020207）；上海市科委资助项目（13510501400）；国家自然科学基金项目（61701295）**作者简介：**王永琦（1977-），女，河南焦作人，副教授，博士研究生，主要研究方向为智能计算与信号处理（wangyongqi17008@163.com）；吴飞（1968-），男，山东人，教授，博士，主要研究方向为计算机并行处理与节能控制；江潇潇（1985-），女，河南洛阳人，讲师，博士，主要研究方向为无限传感器网络、目标跟踪等；王春媛（1983-），女，甘肃白银人，讲师，博士，主要研究方向为信号处理.

机器上的分配及排序，能够快速获得 PMS 问题的近似最优解，但这类求解方法对问题设置具有较严格的限制。近年来，随着智能算法的快速发展，学者们已经将各类智能算法运用到相关 PMS 问题的求解中来，包括遗传算法、禁忌搜索算法、粒子群算法和模拟退火算法等，这类算法能够在满足时间要求的前提下获得 PMS 问题的满意解[6]。然而，上述智能算法存在控制参数多难以选取、收敛速度慢、迭代后期易陷入局部最优等缺点。为此，本文构建了混合教-学算法以快速有效地求解 PMS 问题。

教-学优化算法(teaching-learning based optimization，TLBO)是 Rao 等学者提出的一种新型的群体智能优化算法[7]。该算法通过模拟实际师生授课过程中的“教学”和“学习”两个过程来实现问题的优化，以上两个过程在 TLBO 算法中分别定义为“教学阶段”和“学习阶段”。TLBO 算法具有控制参数少、收敛速度快等优点，并在复杂函数优化、神经网络训练、资源项目调度等多个方面都获得了成功应用[8-11]。同时，鲜有文献将该算法应用于并行机调度问题的研究。

基于以上研究，本文针对加工时间可控的并行机调度，提出了一类考虑拖期与能耗成本优化的调度问题。构建了该问题的整数线性规划模型以利于 CPLEX 求解，并提出了混合教—学求解算法以快速获得中大规模问题的满意解。最后进行了仿真测试，力求验证本文构建的整数线性规划模型和混合优化算法的可行性和有效性。

1　问题模型

1.1　问题描述

假设有 m 台机器负责 n 个订单的加工作业，每个订单有各自的交货期和单位时间拖期惩罚成本，每台机器有多种不同的加工速度，不同加工速度对应的单位时间的能耗成本不同。

本文致力于优化并行机调度问题的拖期与能耗成本总和，需要决策以下三方面的因素：不同机器所负责加工的订单集合；各个机器上订单的加工顺序；各个机器上所有订单的加工速度。

1.2　数学模型

基于以上问题描述，在此定义以下符号以构建本文提出的 PMS-CPT 问题的整数线性规划模型。

m :机器总数,机器编号以 i 表示;

n :订单总数,订单编号以 j 表示;

l :表示机器上的加工位置编号,各个机器均设置 n 个加工位置,即 $1 \le l \le n$;

u_i :表示机器 i 的加工速度总数,速度编号以 s 表示;

ℓ_j :订单 j 的工作负荷;

d_j :订单 j 的交货期;

C_j :订单 j 的完工时间;

T_j :订单 j 的拖期,即 $T_j = \max\{0, C_j - d_j\}$

ω_j :订单 j 单位时间的拖期成本;

S_{is} :机器 i 第 s 个加工速度的取值;

e_{is} :机器 i 以速度 s 运转时产生的单位时间的能耗成本;

t_{jis} :机器 i 以速度 S_{is} 加工订单 j 所需的时间,即 $t_{jis} = \ell_j / S_{is}$

决策变量：

x_{jils} :表示 0-1 变量,当订单 j 在机器 i 上位置 l 处以速度 S_{is} 加工时,取值为 1;否则,为 0;

b_{il} :机器 i 上位置 l 处的订单开始加工的时间。

数学模型：

$$\text{Min} \quad F = \sum_{j=1}^{n} \omega_j \cdot T_j + \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{l=1}^{n}\sum_{s=1}^{u_i} t_{jis} \cdot e_{is} \cdot x_{jils} \tag{1}$$

$$\sum_{i=1}^{m}\sum_{l=1}^{n}\sum_{s=1}^{u_i} x_{jils} = 1, \quad 1 \le j \le n \tag{2}$$

$$\sum_{j=1}^{n}\sum_{s=1}^{u_i} x_{jils} \le 1, 1 \le i \le m, \quad 1 \le l \le n \tag{3}$$

$$b_{i,l+1} - b_{i,l} = \sum_{j=1}^{n}\sum_{s=1}^{u_i} t_{jis} \cdot x_{jils}, \quad 1 \le i \le m, \quad 1 \le l \le n-1 \tag{4}$$

$$T_j \ge b_{i,l} + t_{jis} \cdot x_{jils} - (1 - x_{jils}) \cdot M - d_j, \quad 1 \le i \le m, 1 \le s \le u_i, \quad 1 \le j \le n, 1 \le l \le n \tag{5}$$

$$T_j \ge 0, \quad 1 \le l \le n \tag{6}$$

$$x_{jils} \in \{0,1\}, \quad 1 \le i \le m, \quad 1 \le s \le u_i, \quad 1 \le l \le n, 1 \le j \le n \tag{7}$$

其中，式(1)定义了目标函数，该目标函数包含两项成本：所有订单的拖期惩罚成本总和；所有订单的加工能耗成本总和。约束式(2)表述每一个订单只能安排在一台机器的一处位置上以一个确定的速度进行加工；约束式(3)表示每台机器上的一个位置处至多安排一个订单的加工作业；约束式(4)计算了机器 i 上位置 l 处订单开始加工的时间；约束式(5)(6)用于计算订单 j 的拖期 T_j；式(7)给出了自变量的取值范围。

2　标准教—学优化算法

TLBO 算法利用种群来表示问题的解，当前种群中最优秀的个体称为教师，其余个体称为学生。算法的邻域搜索部分包括“教学阶段”和“学习阶段”两个部分。在教学阶段，TLBO 算法利用教师个体和群体均值个体对学生个体进行教学，进而实现种群的进化；在学习阶段，TLBO 算法从当前群体中随机地选择两个个体，并采用当前个体向较优个体进行学习的策略实现种群的更新。标准 TLBO 算法的实现流程归纳如下：

a)初始化 TLBO 算法的控制参数和待优化问题的参数。包括种群规模学生数目 P，问题变量的维数 D，各个维度自变量的取值范围 $[x_d^l, x_d^u]$。

b)构建初始种群，并计算每个个体的适应度值，个体的初始化公式为 $x_d = x_d^l + rand(0,1) \cdot (x_d^u - x_d^l)$。其中，$rand(0,1)$ 表示 [0,1] 上的随机数。

c)教学阶段。首先，计算当前群体所有个体每一维的均值 $\bar{x}_i$，进而得到平均个体，$\bar{x} = \{\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_D\}$。其次，对于当前种群中的任一个体 x，利用个体更新公式得到新个体 x'。最后，

比较新旧个体 x' 和 x 的适应度值，并采取贪婪保留策略，即保留适应度值较优的个体，放弃适应度值较劣的个体。个体公式为：$x' = x + rand(0,1)\cdot(x^* - \partial \cdot \bar{x})$，其中 x^* 为教师个体，即当前种群中的最优解，教学因子 ∂ 表述随机整数，取值集合为 $\{1,2\}$。

d)学习阶段。首先，随机从经历过教学阶段的种群中选择两个个体，分别记做 x^1 和 x^2；其次，选择 x^1 和 x^2 中较优秀的个体进行学习。以 $f(\cdot)$ 表示适应度函数，假定待优化的问题为最小化问题，个体更新公式为

$$x' = \begin{cases} x + rand(0,1)\cdot(x^1 - x) & 若 f(x^1) < f(x^2) \\ x + rand(0,1)\cdot(x^2 - x) & 否则 \end{cases}$$

e)算法终止条件判断。若满足算法终止条件(达到最大迭代次数或规定的计算时间等)，则终止 TLBO 算法，并输出当前最优解；否则，转 c)。

3　改进教-学算法

3.1　编码与解码

标准 TLBO 算法采用实数编码方式，无法直接适用于离散问题，故需要结合 PMS-CPT 问题的性质在编码与解码方面进行重新的定义和设计。

已知问题参数：机器总数 m，订单总数 n 和每个机器的加工速度总数 u_i ($1 \le i \le m$)，则本文提出的并行机调度问题可采用 $1\times n$ 维数组进行编码表示，各维度编码数值的取值范围为 $[1, m+1)$，编码的第 j 维对应订单 j。第 j 维编码的整数部分对应订单 j 所分配的机器编号；第 j 维编码小数部分第 $1 \sim \lceil \log_{10}^{\max\{u_i|i=1,2,\cdots,m\}} \rceil$ 用于确定订单 j 的加工速度，第 $\lceil \log_{10}^{\max\{u_i|i=1,2,\cdots,m\}} \rceil + 1$ 位至最后用于确定订单 j 在机器上的加工顺序。

解码过程：首先，对编码进行整数取整，以此确定各个订单对应的机器编号。其次，取各维编码小数部分的第 $1 \sim \lceil \log_{10}^{\max\{u_i|i=1,2,\cdots,m\}} \rceil$ 位，并构建各个机器的轮盘赌概率模型以选择各订单的加工速度；针对机器 i，已知该机器的速度类型总数为 u_i，则当相应编码数值处于区间 $[(s-1)/u_i, s/u_i)$ 时，订单的加工速度取值为 S_{is} ($1 \le i \le m$，$1 \le s \le u_i$)。最后，依据分配到同一机器上的订单对应的编码数值的第 $\lceil \log_{10}^{\max\{u_i|i=1,2,\cdots,m\}} \rceil + 1$ 位至最后的数值大小，进行升序排列以确定订单加工顺序。

图 1 给出了编码与解码示意图，算例参数为：机器总数 m=2，订单总数 n=5，每个机器的加工速度种类 u_i=3 ($1 \le i \le m$)。因此，编码的各个维度的取值范围为 $[1,3)$，编码的整数部分表示订单-机器分配情况，编码的小数部分第一位用于确定订单加工速度，小数部分第二位至最后用于确定订单加工顺序。已知初始编码{2.629, 2.812, 1.254, 2.827, 2.265, 1.195}。解码结果归纳如下：

订单 1 在机器 2 上以速度 S_{22} 进行加工；

订单 2 在机器 2 上以速度 S_{23} 进行加工；

订单 3 在机器 1 上以速度 S_{11} 进行加工；

订单 4 在机器 2 上以速度 S_{23} 进行加工；

订单 5 在机器 1 上以速度 S_{11} 进行加工；

机器 1 的订单加工顺序为{3,5}；

机器 2 的订单加工顺序为{2,4,1}。

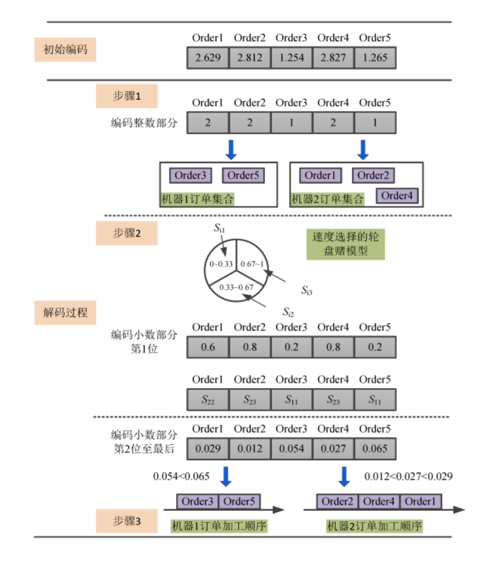


图 1　编码与解码示意图

3.2　基于 VNS 的局部搜索算子

与其他群智能优化算法相类似，标准 TLBO 算法存在着迭代后期易陷入局部最优、寻优性能不足等缺点。为此，本文构建了基于变邻域搜索(Variable neighborhood search，VNS)的局部搜索算子，力求增强标准 TLBO 算法的搜索性能。

VNS 算法通过系统地改变邻域结构集以拓展当前算法的搜索范围，进而获得局部最优解；同时，基于此局部最优解，再次系统地改变邻域结构集来进一步拓展搜索的范围，力求找到另一个局部最优解[12]。下图给出了 VNS 算法搜索过程的示意图。

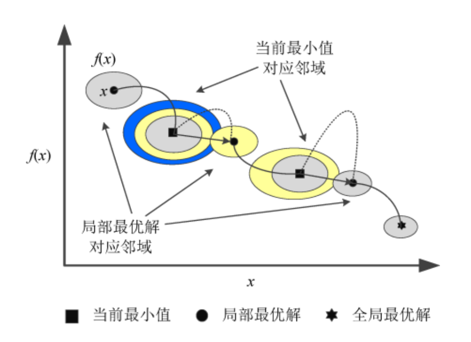


图 2　VNS 算法示意图

给定解 $x = \{x_1, x_2, \cdots, x_D\}$，VNS 算法的核心在于确定 x 的邻域结构和搜索流程。给定循环次数 K，本文构建的基于 VNS 的局部搜索算子可表述为：

a)令 $k=1$，转 b)。

b)令 $d=1$，转 c)。

c)若 $d \le D$，转 d)；否则，转 f)。

d)令 $x'_d = x_d + sign \cdot rand(0,1) \cdot \beta \cdot m$，转 e)。

e)令 $d \leftarrow d+1$，转 c)。

f)计算 x' 的适应度值 $f(x')$，若 $f(x') \le f(x)$，则令 $x \leftarrow x'$，转 g)。

g)令 $k \leftarrow k+1$，若 $k < K$，转 b)；否则，结束 VNS 搜索。其中，d)定义了邻域变换公式，$sign$ 表述随机整数，取值集合为 $\{-1,1\}$，β 为邻域控制参数取值为 $(0,1]$，m 为模型参数(即机器总数)。

3.3 改进教-学算法流程

结合以上描述，给出求解 PMS-CPT 问题的 HTLBO 算法流程，归纳如下：

a)初始化算法的各个参数，包括种群规模、算法终止条件等。

b)采用随机方法构建初始种群，并计算适应度值。

c)计算当前种群的平均个体，并利用“教学阶段”的邻域更新公式更新种群。

d)利用“学习阶段”的邻域更新公式更新种群。

e)针对“教学阶段”和“学习阶段”个体更新失败的解，采用基于 VNS 的局部搜索算子进行个体的二次更新操作。

f)若算法满足终止条件，则终止迭代搜索并输出当前最优解；否则，转 c)。

4　仿真实验及分析

4.1　算例设计

由于难以从现有的文献中获得与本文提出的并行机调度问题类似的测试数据集，因此，本文依据 Hall 和 Posner 学者提出的机器调度问题测试数据生成方法构建了本文的测试算例[13]。

本文构建的测试算例分为：小规模算例和中大规模算例，共计 21 个。其中，小规模测试算例能够通过 CPLEX 进行求解，并将所得的最优解用于混合教-学算法的性能分析；中大规模算例则无法在合理的计算时间内借助 CPLEX 获得精确解，因此采用其他同类型智能算法所得的满意解进行混合教-学算法的性能分析。对于小规模算例，机器数目 $m \in \{3,4,5\}$，订单数目 $n \in \{5,10,15\}$；对于中大规模算例，机器数目 $m \in \{3,4,5\}$，订单数目 $n \in \{30,50,80,100\}$。对于算例 (m,n)，相关参数取值如下：

a)订单 j 的工作负荷 ℓ_j 在均匀分布 $U(1,100)$ 上产生；订单 j 单位时间的拖期成本系数 ω_j 在均匀分布 $U(0,1)$ 上产生；

b)各机器加工速度类型总数 u_i=5，机器 i 第 s 个加工速度的取值 S_{is} 在均匀分布 $U(5,10)$ 上产生；

c)机器 i 以速度 s 运转时产生的单位时间的能耗成本 e_{is} 在均匀分布 $U(0,1)$ 上产生，且加工速度越低，对应的 e_{is} 值越大；

d)订单 j 的交货期 d_j 在均匀分布 $U(0.5\cdot\bar{D}, 2\cdot\bar{D})$ 上产生，其中 $\bar{D}$ 的表达式为

$$\bar{D} = 0.5 \cdot \frac{\sum_{j=1}^{n} l_j}{m} \div \frac{\sum_{i=1}^{m}\sum_{s=1}^{u_i} S_{is}}{\sum_{i=1}^{m} u_i}$$

在 MATLAB2016a 平台上进行编程测试，计算机参数为主频 1.6 GHz、内存 8 GB、Intel Core i5-8250UCPU，PMS-CPT 问题的整数线性规划通过 CPLEX-MATLAB 接口借助 CPLEX 12.4 进行求解。针对各个测试算例，HTLBO 算法进行 30 次独立仿真测试。在小规模算例中，HTLBO 算法的种群规模设为 30，最大计算时间设为 10s，参数 K 取值为 5，参数 β 取值为 0.2；在中大规模算例中，HTLBO 算法的种群规模设为 80，最大计算时间设为 150s，参数 K 取值为 15，参数 β 取值为 0.1。

4.2　测试结果

首先，利用 CPLEX 和 HTLBO 算法对小规模测试算例进行了求解，测试结果见表 1。其中，F^* 表示 CPLEX 所得的精确解，后一列为 CPLEX 求解各个算例所需的计算时间；F_b 和 F_m 分别表示 HTLBO 算法在 30 次独立仿真测试中所获得的最优解和最劣解，P_b 表示获得最优解的实验次数占总实验次数（即 30 次）的百分比。

由表 1 的测试结果可知，CPLEX 能够利用本文构建的整数线性规划模型获得 PMS-CPT 调度问题的精确解(F^*)，据此验证了整数线性规划模型的有效性。但 CPLEX 所消耗的计算时间较大，算例(5,15)达到了 2815s。对于各个小规模测试算例 (m,n)，本文构建的 HTLBO 能够获得问题的准确解(即 F_b=F^*)，且 30 次独立实验所得准确解的百分比(P_b)不低于 83.74%；同时，最劣解(F_m)与准确解数值较为接近。考虑到 HTLBO 的计算时间仅为 10s，远小于 CPLEX 的计算时间，据此可验证 HTLBO 算法在解决小规模算例时的有效性。

表 1　小规模算例测试结果

(m,n)	CPLEX		HTLBO		
	F^*	时间/s	F_b	F_w	P_b/%
(3,5)	20.46	233	20.46	21.05	94.68
(3,10)	38.52	412	38.52	39.44	93.16
(3,15)	111.45	1686	111.45	113.21	91.16
(4,5)	10.67	351	10.67	10.93	95.01
(4,10)	28.14	1514	28.14	28.94	91.53
(4,15)	87.52	1793	87.52	88.72	88.15
(5,5)	13.35	564	13.35	13.80	91.80
(5,10)	38.94	2382	38.94	39.44	88.09
(5,15)	63.03	2815	63.03	63.69	83.74

进一步，进行了中大规模算例的测试。考虑到 CPLEX 无法在合理的时间内获得这类测试算例的准确解，因此将 HTLBO 算法与 TLBO 算法和粒子群算法(particle swarm optimization，PSO)算法[14]进行比较。针对每个算例，各算法运行 30 次，表 2 统计了寻优结果的平均值($\bar{F}$)与标准差(σ)。此外，变量 κ 表

示相对平均偏差百分比，针对各个测试算例 (m,n)，令三个算法共计 90 测仿真所得的最优解为 F'，变量 κ 的计算公式为 $(\bar{F}-F')/F'\times100\%$。

由表 2 的测试结果可知，融合了基于 VNS 的局部搜索算子的 HTLBO 算法在均值($\bar{F}$)和相对平均偏差百分比(κ)两个指标方面均要优于 TLBO 算法和 PSO 算法，据此可知 HTLBO 算法的搜索结果总体较其他两种算法更优秀。同时，HTLBO 算法30次独立仿真测试结果的标准差(σ 值)更小，这说明HTLBO 算法具有较强的鲁棒性，表现更为稳定。

5　结束语

本文针对加工时间可控的并行机调度，研究了一类考虑拖期与能耗成本优化的调度问题。构建了该问题的整数线性规划模型，并借助 CPLEX 对小规模问题进行精确求解。同时，设计了混合教-学算法以快速获得中大规模问题的满意解。对 21 个测试算法进行了仿真测试，实验结果表明 CPLEX 和混合教—学算法能够获得小规模问题的精确解，且混合教—学算法的计算耗时较短；同时相较于标准 TLBO 算法和 PSO 算法，混合教—学算法在求解中大规模测试算例时表现出更佳的性能，且算法鲁棒性更强。

表 2　中大规模算例测试结果

(m,n)	HTLBO			TLBO			PSO		
	$\bar{F}$	σ	κ/%	$\bar{F}$	σ	κ/%	$\bar{F}$	σ	κ/%
(3,30)	337.58	2.67	3.49	340.93	3.58	4.51	338.77	3.75	3.85
(3,50)	812.57	7.66	4.21	819.78	12.50	5.13	817.76	10.32	4.87
(3,80)	2354.43	24.10	5.35	2369.27	37.40	6.01	2360.39	32.28	5.61
(3,100)	3138.57	34.17	6.30	3157.63	56.06	6.95	3149.36	43.80	6.67
(4,30)	318.13	2.91	3.62	321.23	5.48	4.63	320.17	4.98	4.29
(4,50)	625.26	7.61	4.02	632.52	12.18	5.23	629.51	10.23	4.73
(4,80)	1679.45	12.09	5.09	1686.75	24.49	5.55	1683.92	20.81	5.37
(4,100)	2183.21	16.46	6.66	2197.24	30.08	7.35	2192.75	34.46	7.13
(5,30)	223.67	1.89	3.94	226.73	2.97	5.37	225.17	2.69	4.64
(5,50)	450.66	3.89	4.47	457.76	7.16	6.12	454.09	6.93	5.27
(5,80)	1142.95	12.50	5.79	1152.99	24.22	6.72	1145.40	20.08	6.01
(5,100)	2208.56	20.45	6.72	2225.40	34.38	7.75	2217.73	31.27	7.62

参考文献：

[1] Cheng T C E, Sin C C S. A state-of-the-art review of parallel-machine scheduling research [J]. European Journal of Operational Research, 1990, 47 (3): 271-292.

[2] Adamu M O, Abass O. Parallel machine scheduling to maximize the weighted number of just-in-time jobs [J]. Journal of Applied Science and Technology, 2010, 15 (1-2) .

[3] Yin Y, Cheng S R, Cheng T C E, et al. Just-in-time scheduling with two competing agents on unrelated parallel machines [J]. Omega, 2016, 63: 41-47.

[4] Li K, Zhang X, Leung J Y T, et al. Parallel machine scheduling problems in green manufacturing industry [J]. Journal of Manufacturing Systems, 2016, 38: 98-106.

[5] Lin Y K, Pfund M E, Fowler J W. Heuristics for minimizing regular performance measures in unrelated parallel machine scheduling problems [J]. Computers & Operations Research, 2011, 38 (6): 901-916.

[6] Balin S. Non-identical parallel machine scheduling using genetic algorithm [J]. Expert Systems with Applications, 2011, 38 (6): 6814-6821.

[7] Rao R V, Savsani V J, Vakharia D P. Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems [J]. Information Sciences, 2012, 183 (1): 1-15.

[8] Rao R V, Patel V. An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems [J]. Scientia Iranica, 2013, 20 (3): 710-720.

[9] 高立群，欧阳海滨，孔祥勇，等. 带有交叉操作的教-学优化算法 [J]. 东北大学学报：自然科学版, 2014, 35 (3): 323-327.

[10] 于坤杰，王昕，王振雷. 基于反馈的精英教学优化算法 [J]. 自动化学报, 2014, 40 (9): 1976-1983.

[11] 拓守恒，雍龙泉，邓方安. “教与学”优化算法研究综述 [J]. 计算机应用研究, 2013, 30 (7): 1933-1938.

[12] Hemmelmayr V C, Doerner K F, Hartl R F. A variable neighborhood search heuristic for periodic routing problems [J]. European Journal of Operational Research, 2009, 195 (3): 791-802.

[13] Hall N G, Posner M E. Generating experimental data for computational testing with machine scheduling applications [J]. Operations Research, 2001, 49 (6): 854-865.

[14] AlRashidi M R, El-Hawary M E. A survey of particle swarm optimization applications in electric power systems [J]. IEEE Trans on Evolutionary Computation, 2009, 13 (4): 913-918.